fig. 1A

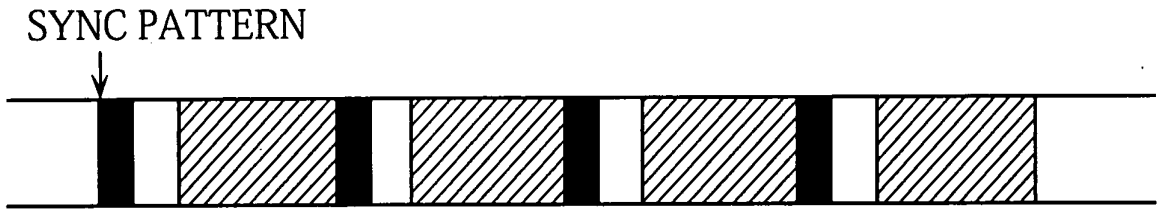

fig. 1B

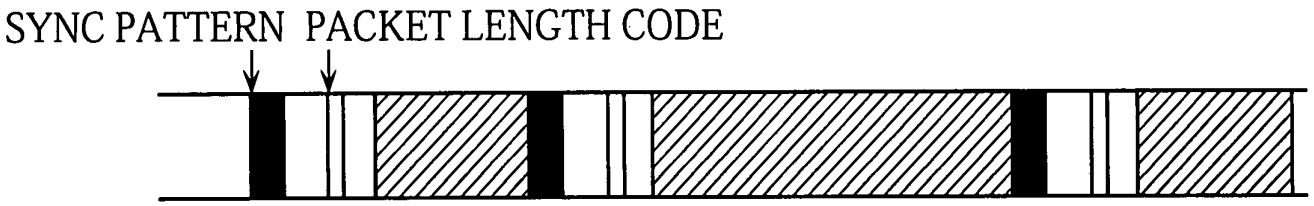

fig. 1C

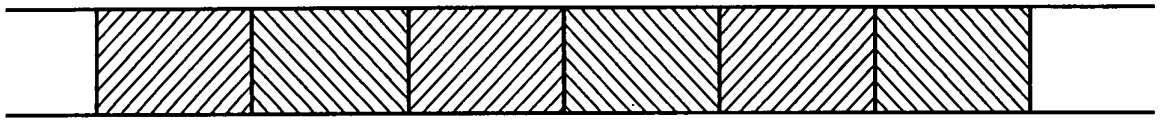

fig. 2

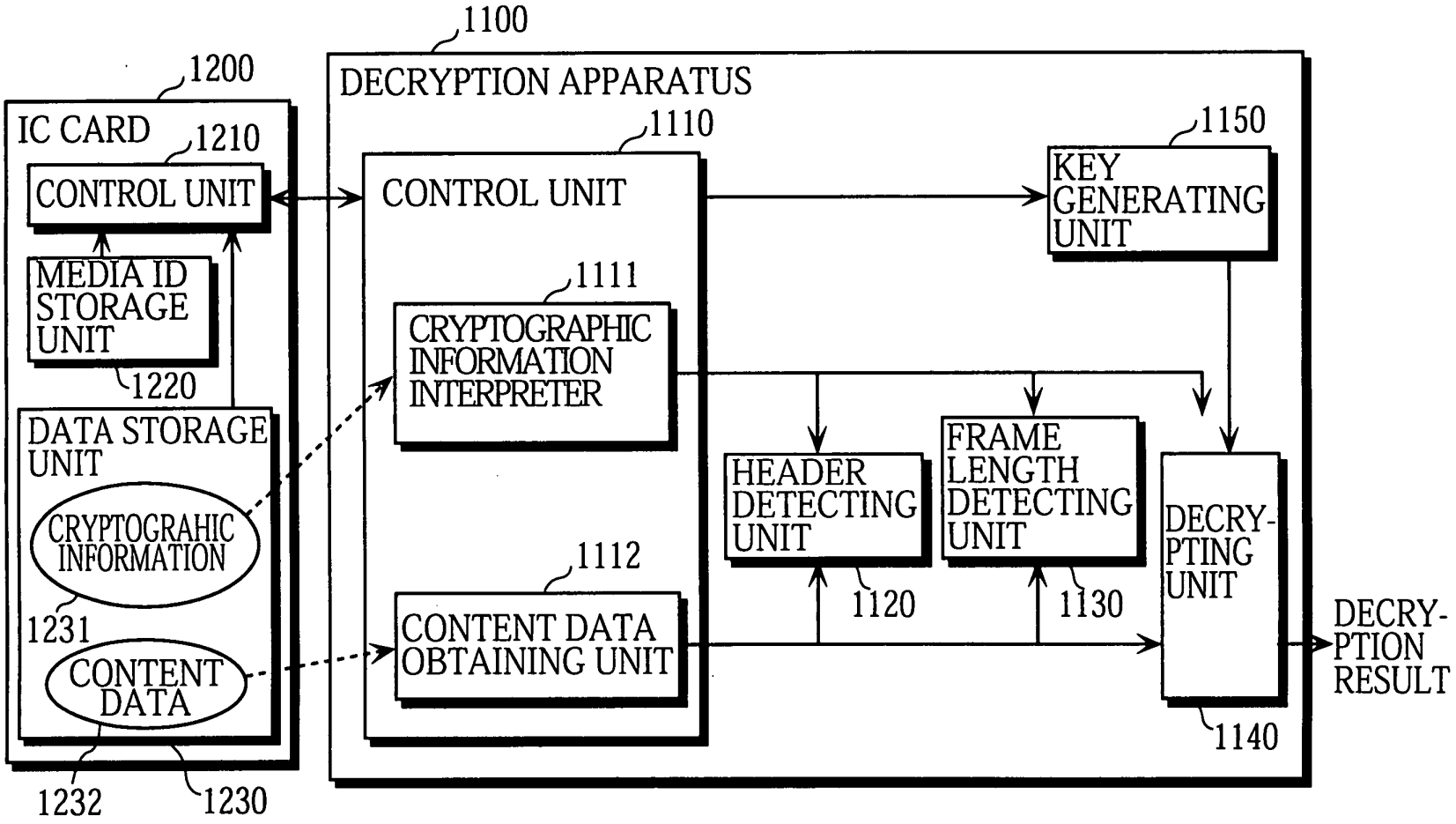

fig. 3

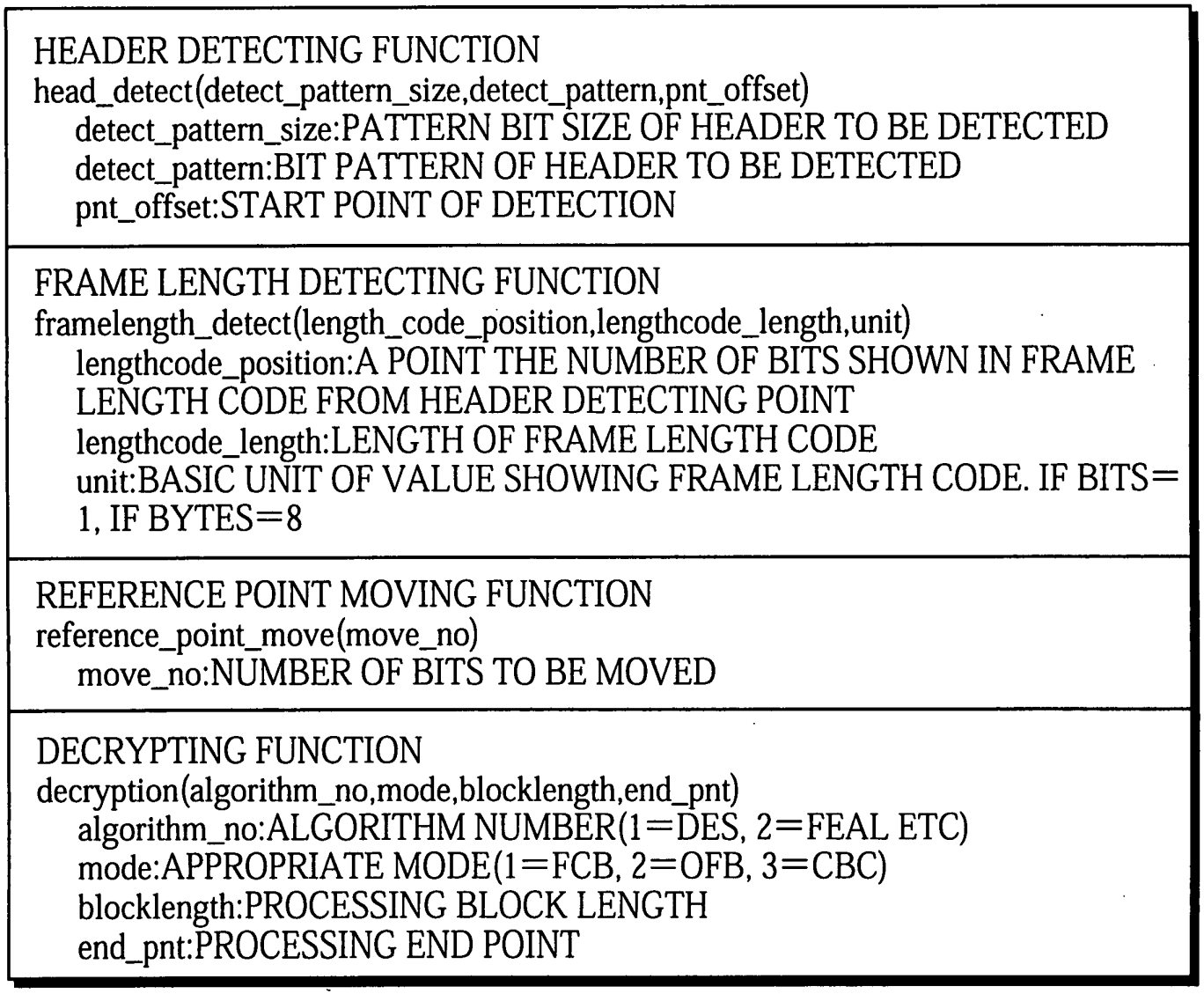
HEADER DETECTING FUNCTION
head_detect(detect_pattern_size,detect_pattern,pnt_offset)
detect_pattern_size:PATTERN BIT SIZE OF HEADER TO BE DETECTED
detect_pattern:BIT PATTERN OF HEADER TO BE DETECTED
pnt_offset:START POINT OF DETECTION
FRAME LENGTH DETECTING FUNCTION
framelength_detect(length_code_position,lengthcode_length,unit)
lengthcode_position:A POINT THE NUMBER OF BITS SHOWN IN FRAME LENGTH CODE FROM HEADER DETECTING POINT
lengthcode_length:LENGTH OF FRAME LENGTH CODE
unit:BASIC UNIT OF VALUE SHOWING FRAME LENGTH CODE. IF BITS=1, IF BYTES=8
REFERENCE POINT MOVING FUNCTION
reference_point_move(move_no)
move_no:NUMBER OF BITS TO BE MOVED
DECRYPTING FUNCTION
decryption(algorithm_no,mode,blocklength,end_pnt)
algorithm_no:ALGORITHM NUMBER(1=DES, 2=FEAL ETC)
mode:APPROPRIATE MODE(1=FCB, 2=OFB, 3=CBC)
blocklength:PROCESSING BLOCK LENGTH
end_pnt:PROCESSING END POINT

fig. 4

```
ref_pnt=0;
while(endofdata)
{
        pnt_offset=ref_pnt;
        ref_pnt=head_detect(12,0xfff,ref_pnt);
        frame_length=frame_length_detect(31,13,8);
        ref_pnt=reference_point_move(13);
        end_pnt=pnt_offset+framelength;
        decryption(1,3,64,end_pnt);
        ref_pnt=end_pnt;
}
end
```

fig. 5

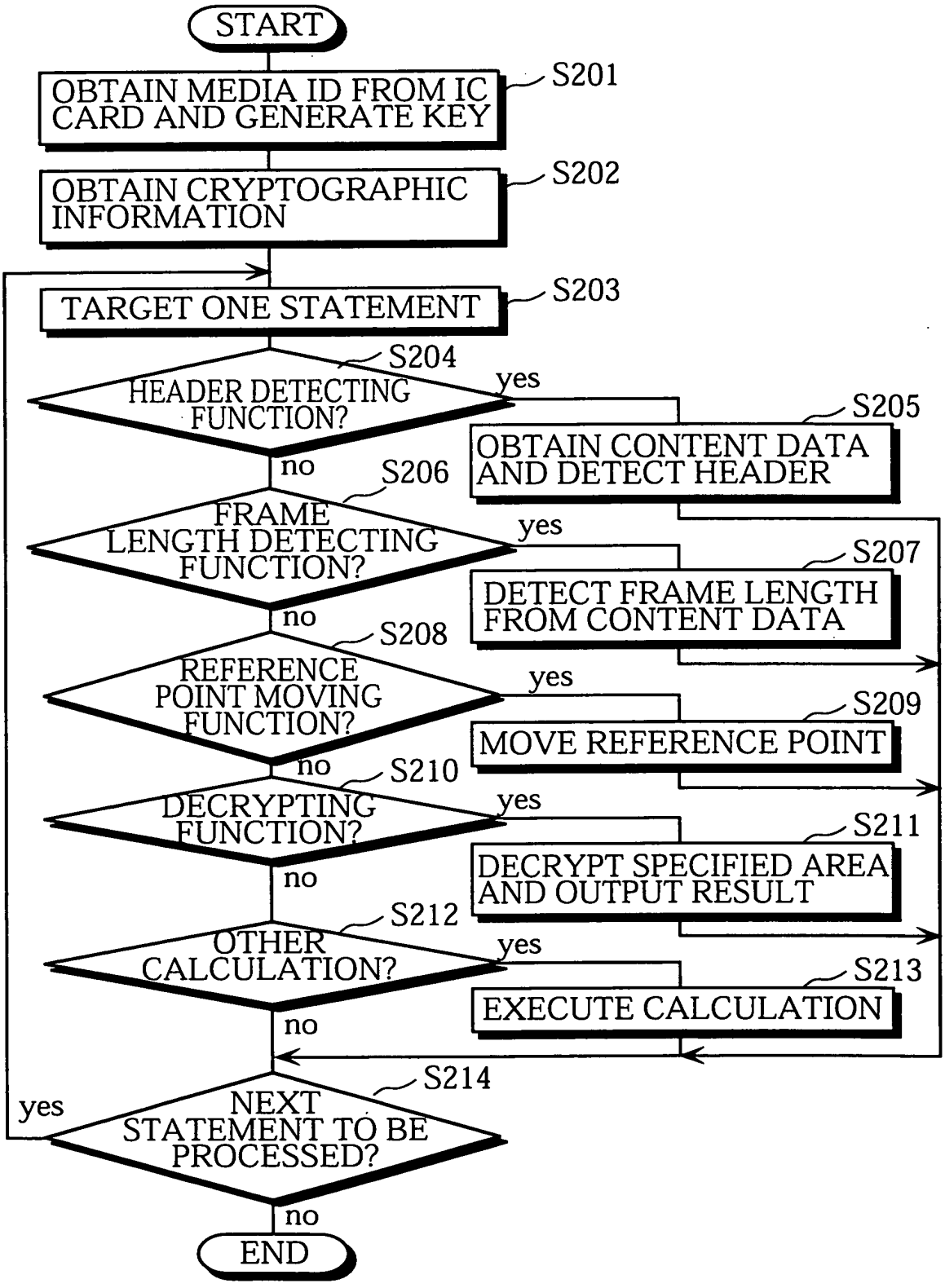
START
OBTAIN MEDIA ID FROM IC CARD AND GENERATE KEY
S201
OBTAIN CRYPTOGRAPHIC INFORMATION
S202
TARGET ONE STATEMENT
S203
HEADER DETECTING FUNCTION?
S204
yes
OBTAIN CONTENT DATA AND DETECT HEADER
S205
no
FRAME LENGTH DETECTING FUNCTION?
S206
yes
DETECT FRAME LENGTH FROM CONTENT DATA
S207
no
REFERENCE POINT MOVING FUNCTION?
S208
yes
MOVE REFERENCE POINT
S209
no
DECRYPTING FUNCTION?
S210
yes
DECRYPT SPECIFIED AREA AND OUTPUT RESULT
S211
no
OTHER CALCULATION?
S212
yes
EXECUTE CALCULATION
S213
no
NEXT STATEMENT TO BE PROCESSED?
S214
yes
no
END

fig. 6

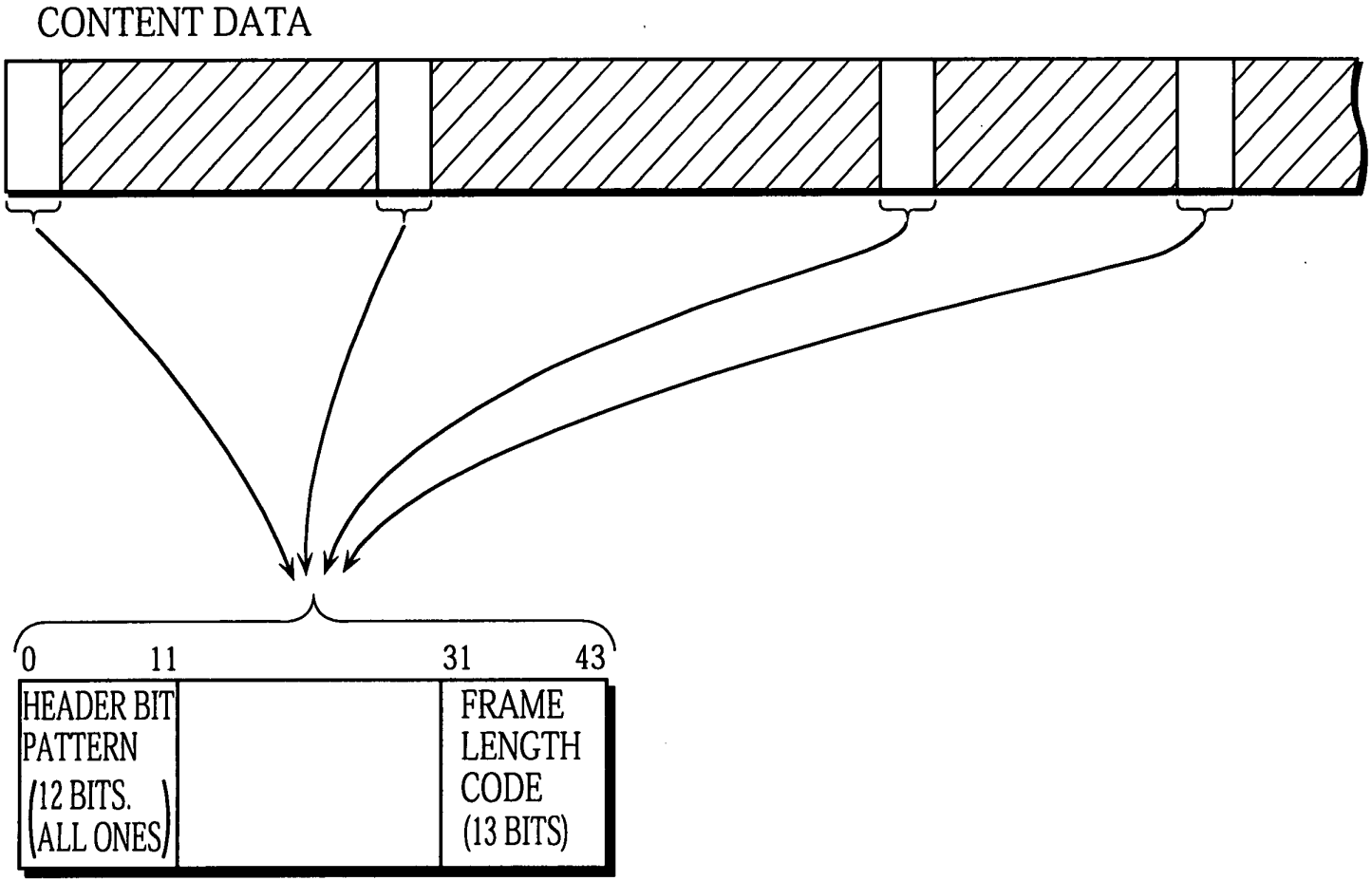
CONTENT DATA
0
11
31
43
HEADER BIT PATTERN (12 BITS. ALL ONES)
FRAME LENGTH CODE (13 BITS)

fig. 7

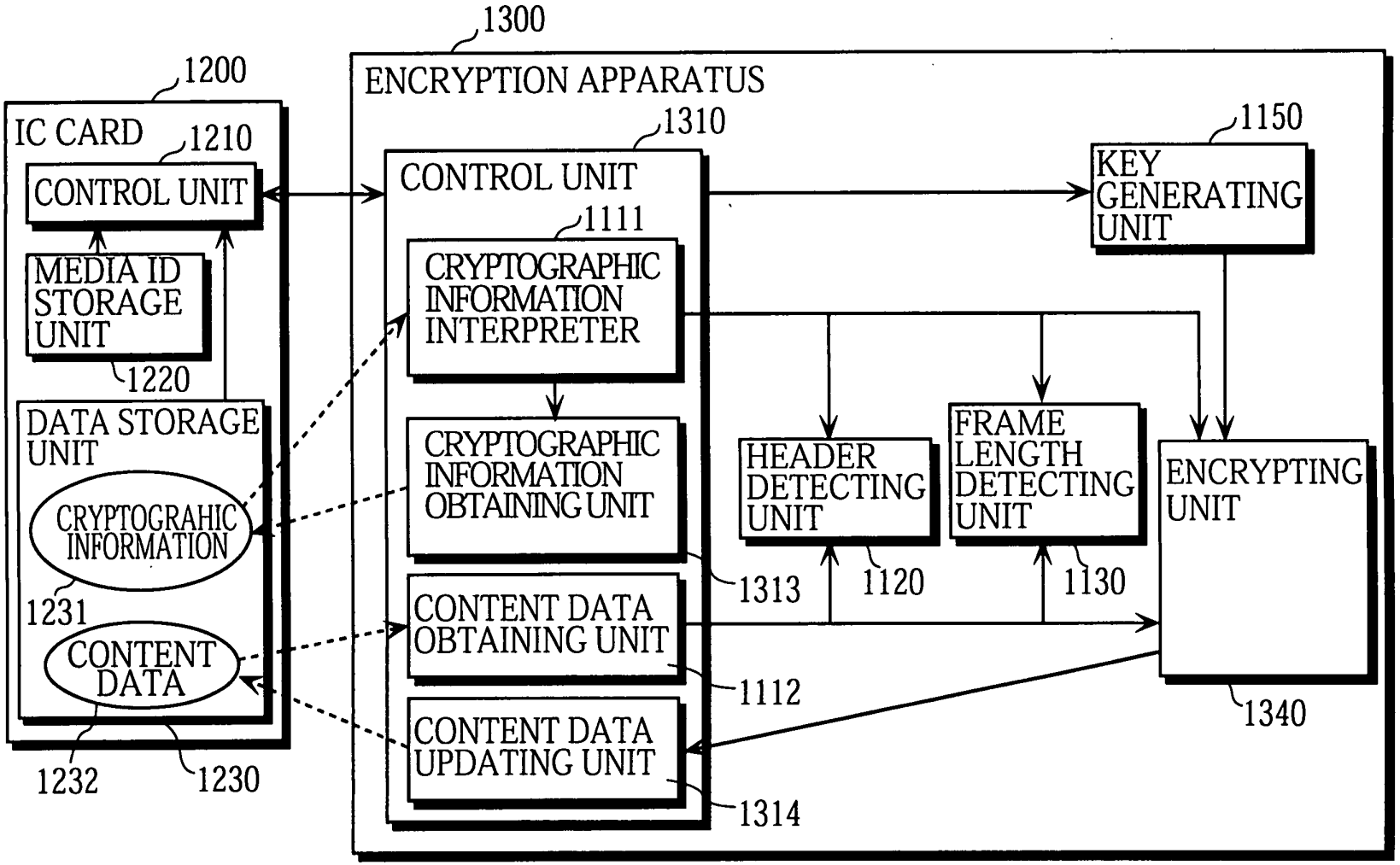
1300
ENCRYPTION APPARATUS
1200
IC CARD
1210
CONTROL UNIT
MEDIA ID STORAGE UNIT
1220
DATA STORAGE UNIT
CRYPTOGRAHIC INFORMATION
1231
CONTENT DATA
1232
1230
1310
CONTROL UNIT
1111
CRYPTOGRAPHIC INFORMATION INTERPRETER
CRYPTOGRAPHIC INFORMATION OBTAINING UNIT
CONTENT DATA OBTAINING UNIT
1112
CONTENT DATA UPDATING UNIT
1314
1313
1150
KEY GENERATING UNIT
HEADER DETECTING UNIT
1120
FRAME LENGTH DETECTING UNIT
1130
ENCRYPTING UNIT
1340

fig. 8

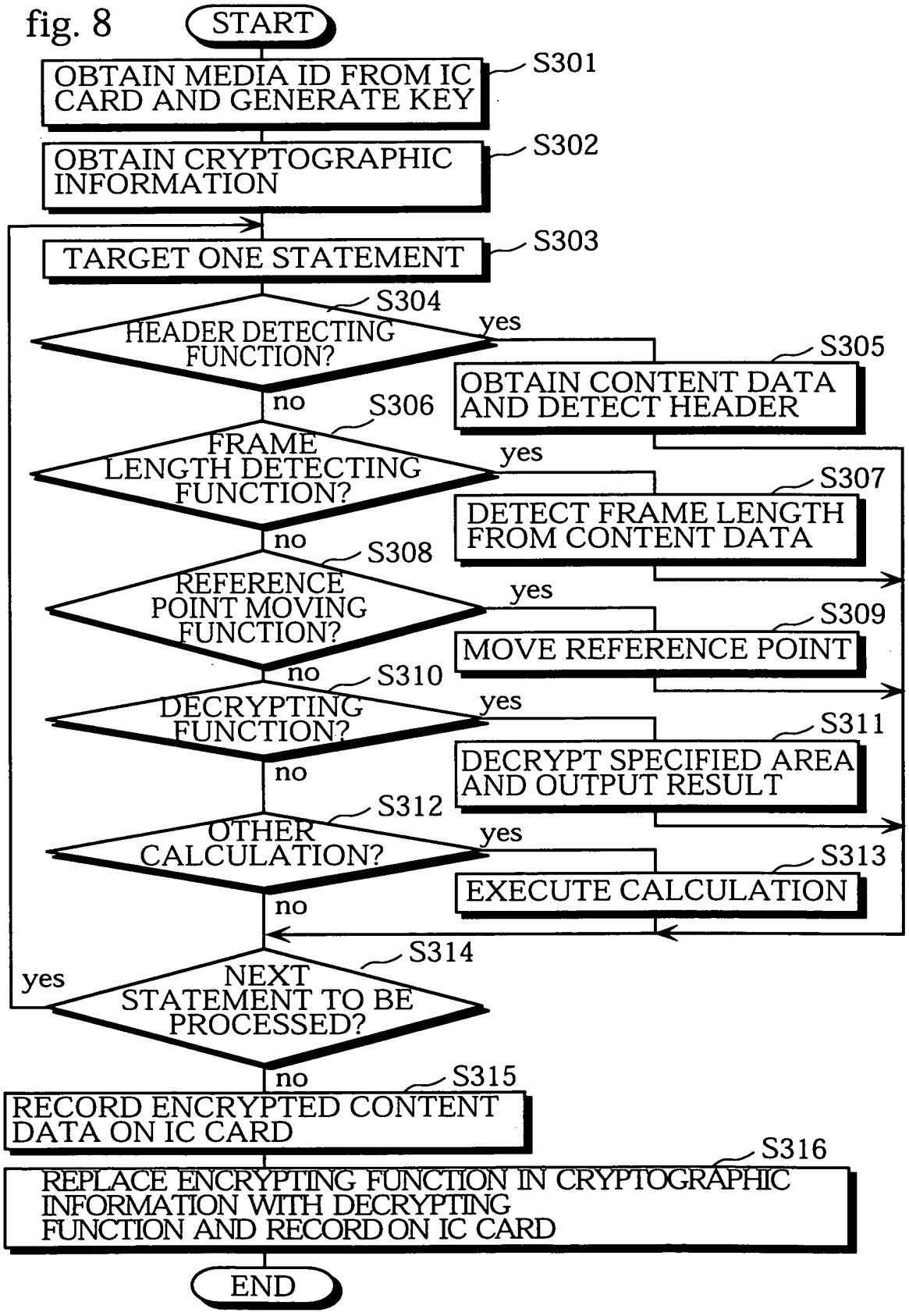
START
OBTAIN MEDIA ID FROM IC CARD AND GENERATE KEY
S301
OBTAIN CRYPTOGRAPHIC INFORMATION
S302
TARGET ONE STATEMENT
S303
HEADER DETECTING FUNCTION?
S304
yes
OBTAIN CONTENT DATA AND DETECT HEADER
S305
no
FRAME LENGTH DETECTING FUNCTION?
S306
yes
DETECT FRAME LENGTH FROM CONTENT DATA
S307
no
REFERENCE POINT MOVING FUNCTION?
S308
yes
MOVE REFERENCE POINT
S309
no
DECRYPTING FUNCTION?
S310
yes
DECRYPT SPECIFIED AREA AND OUTPUT RESULT
S311
no
OTHER CALCULATION?
S312
yes
EXECUTE CALCULATION
S313
no
NEXT STATEMENT TO BE PROCESSED?
S314
yes
no
RECORD ENCRYPTED CONTENT DATA ON IC CARD
S315
REPLACE ENCRYPTING FUNCTION IN CRYPTOGRAPHIC INFORMATION WITH DECRYPTING FUNCTION AND RECORD ON IC CARD
S316
END

fig. 9

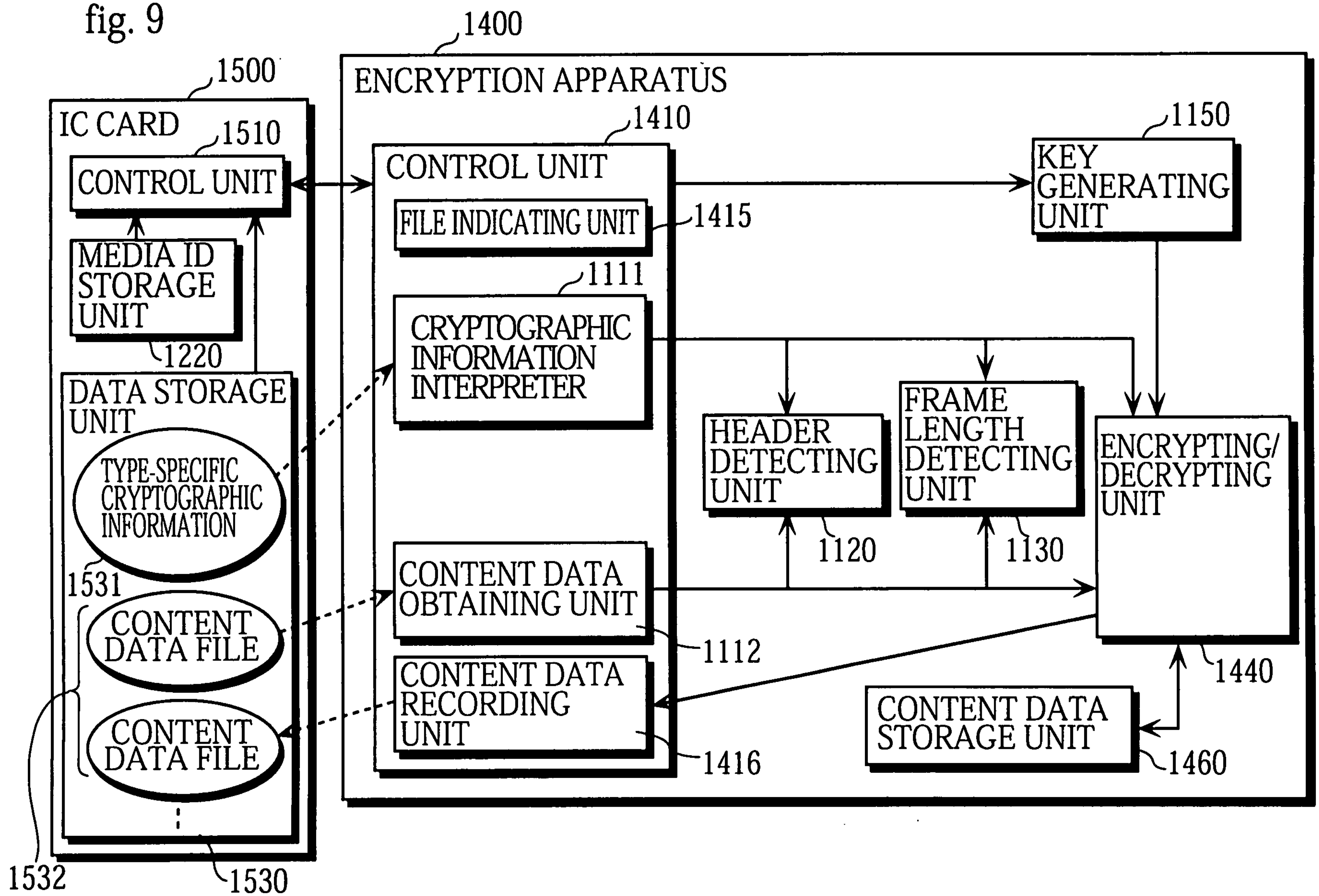
1400
ENCRYPTION APPARATUS
1410
CONTROL UNIT
FILE INDICATING UNIT
1415
1111
CRYPTOGRAPHIC INFORMATION INTERPRETER
CONTENT DATA OBTAINING UNIT
1112
CONTENT DATA RECORDING UNIT
1416
1150
KEY GENERATING UNIT
HEADER DETECTING UNIT
1120
FRAME LENGTH DETECTING UNIT
1130
ENCRYPTING/ DECRYPTING UNIT
1440
CONTENT DATA STORAGE UNIT
1460
1500
IC CARD
1510
CONTROL UNIT
MEDIA ID STORAGE UNIT
1220
DATA STORAGE UNIT
TYPE-SPECIFIC CRYPTOGRAPHIC INFORMATION
1531
CONTENT DATA FILE
CONTENT DATA FILE
1532
1530

fig. 10

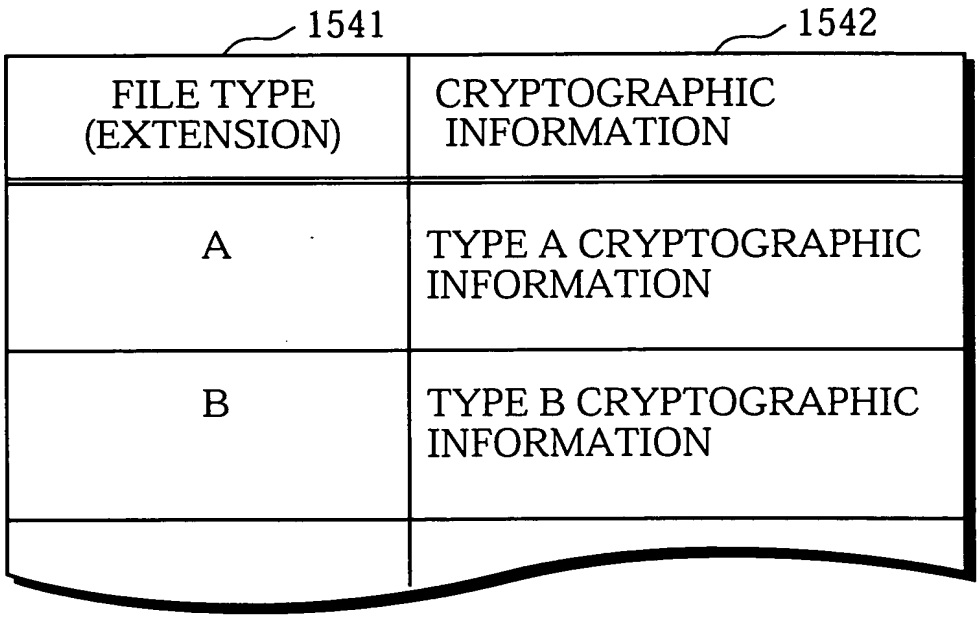
1541
1542
FILE TYPE (EXTENSION)
CRYPTOGRAPHIC INFORMATION
A
TYPE A CRYPTOGRAPHIC INFORMATION
B
TYPE B CRYPTOGRAPHIC INFORMATION

fig. 11

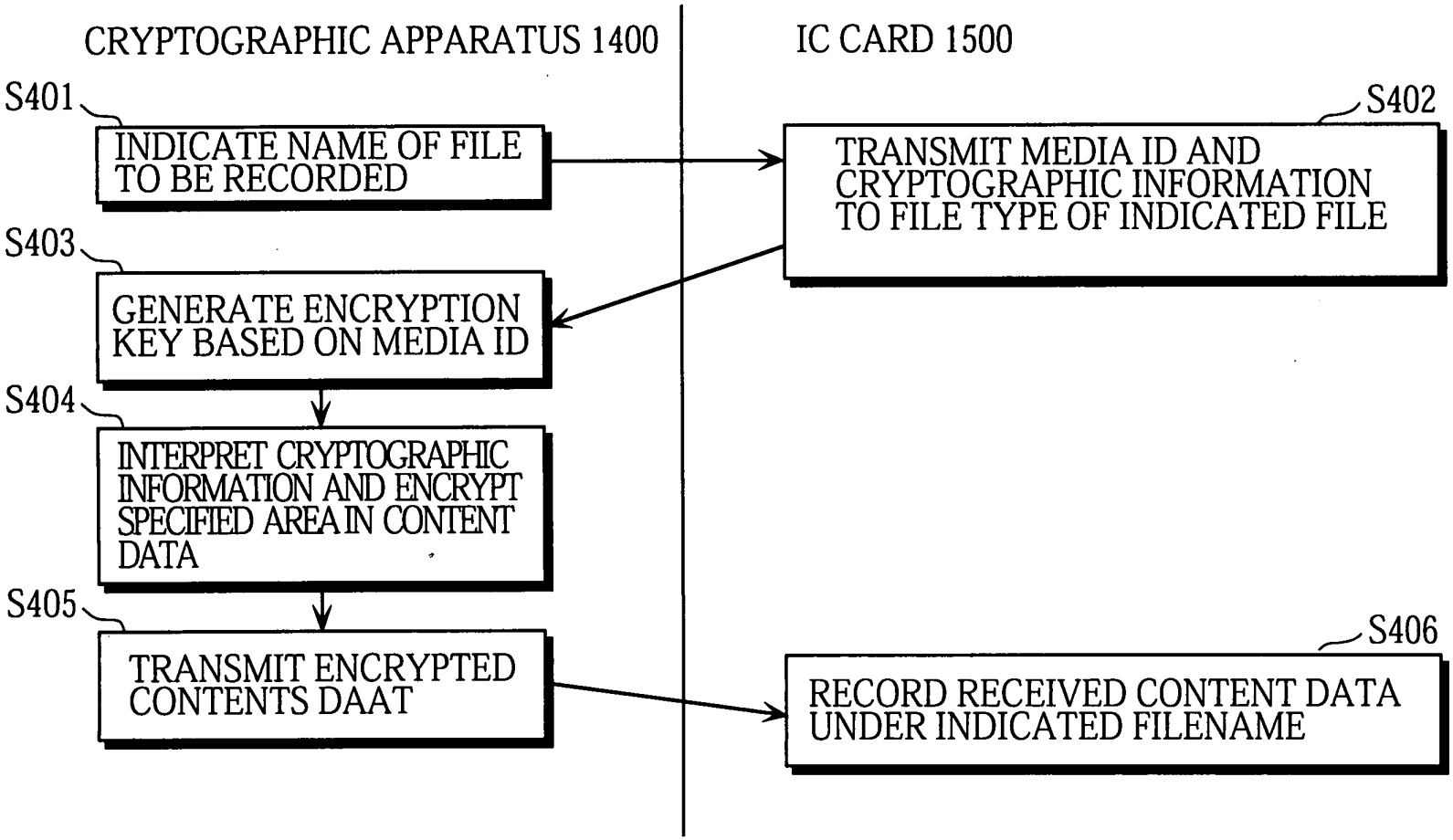
CRYPTOGRAPHIC APPARATUS 1400
IC CARD 1500
S401
INDICATE NAME OF FILE TO BE RECORDED
S402
TRANSMIT MEDIA ID AND CRYPTOGRAPHIC INFORMATION TO FILE TYPE OF INDICATED FILE
S403
GENERATE ENCRYPTION KEY BASED ON MEDIA ID
S404
INTERPRET CRYPTOGRAPHIC INFORMATION AND ENCRYPT SPECIFIED AREA IN CONTENT DATA
S405
TRANSMIT ENCRYPTED CONTENTS DAAT
S406
RECORD RECEIVED CONTENT DATA UNDER INDICATED FILENAME

fig. 12

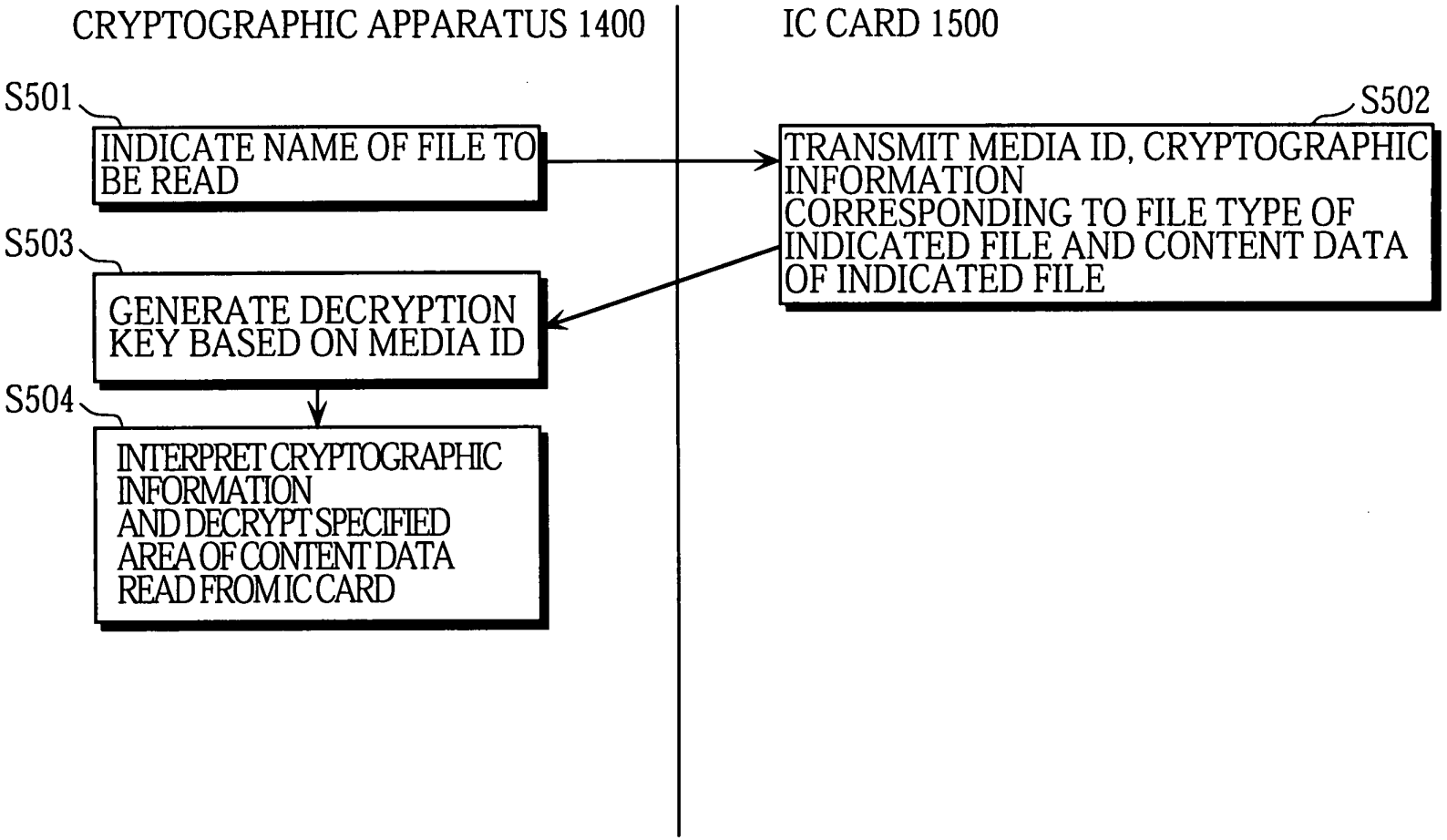
CRYPTOGRAPHIC APPARATUS 1400
IC CARD 1500
S501
INDICATE NAME OF FILE TO BE READ
S502
TRANSMIT MEDIA ID, CRYPTOGRAPHIC INFORMATION CORRESPONDING TO FILE TYPE OF INDICATED FILE AND CONTENT DATA OF INDICATED FILE
S503
GENERATE DECRYPTION KEY BASED ON MEDIA ID
S504
INTERPRET CRYPTOGRAPHIC INFORMATION AND DECRYPT SPECIFIED AREA OF CONTENT DATA READ FROM IC CARD

fig. 13

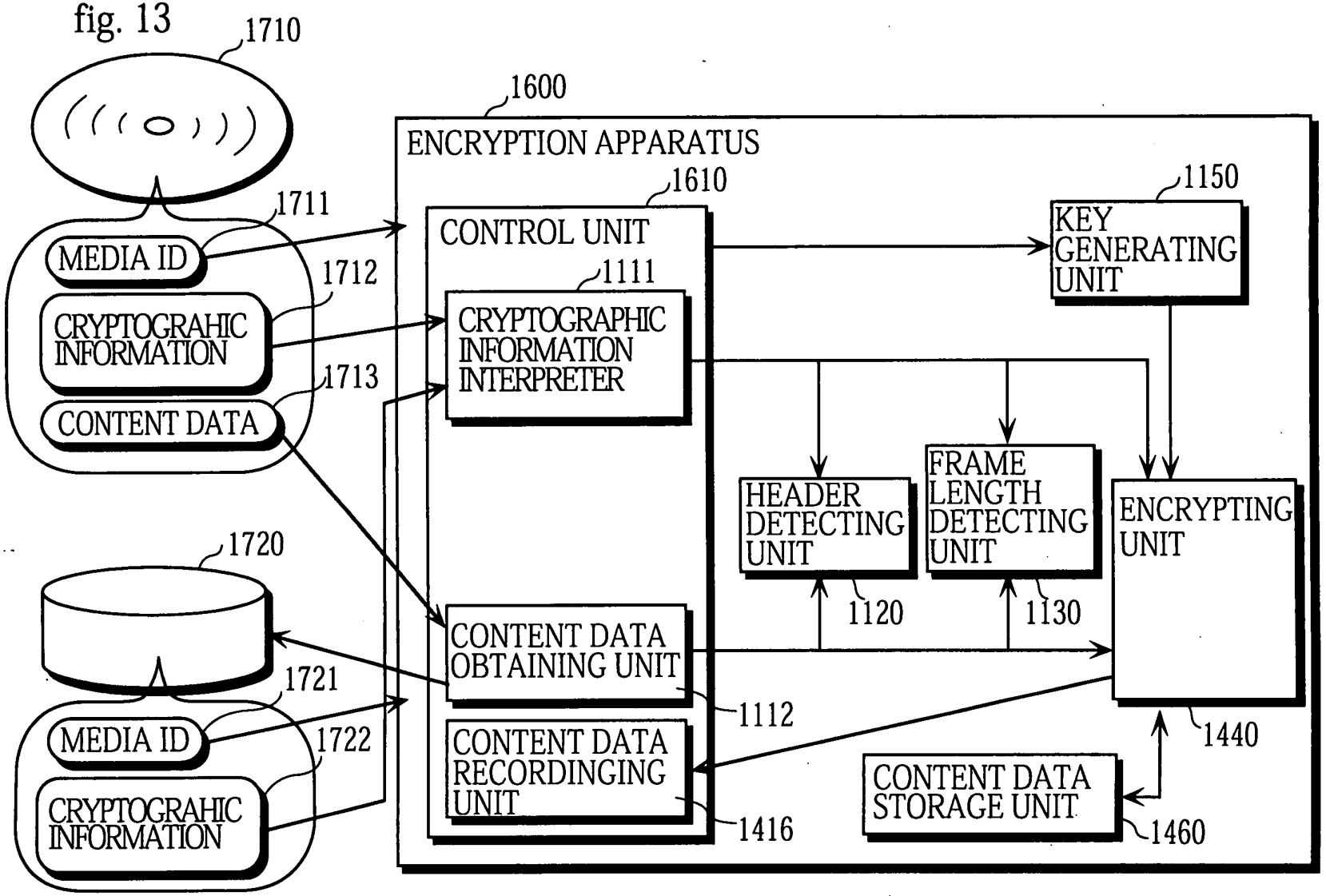

fig. 14

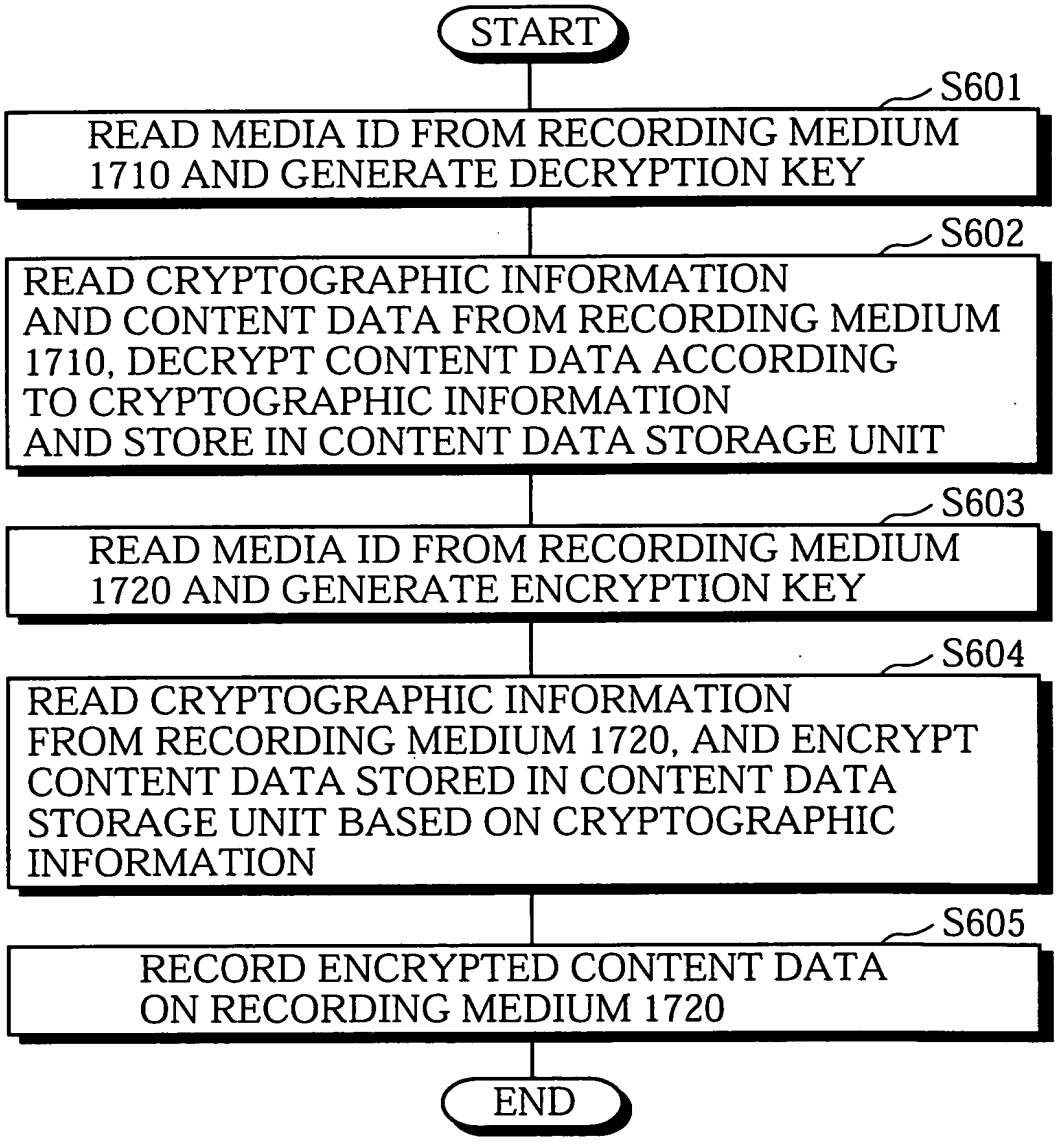
START
S601
READ MEDIA ID FROM RECORDING MEDIUM 1710 AND GENERATE DECRYPTION KEY
S602
READ CRYPTOGRAPHIC INFORMATION AND CONTENT DATA FROM RECORDING MEDIUM 1710, DECRYPT CONTENT DATA ACCORDING TO CRYPTOGRAPHIC INFORMATION AND STORE IN CONTENT DATA STORAGE UNIT
S603
READ MEDIA ID FROM RECORDING MEDIUM 1720 AND GENERATE ENCRYPTION KEY
S604
READ CRYPTOGRAPHIC INFORMATION FROM RECORDING MEDIUM 1720, AND ENCRYPT CONTENT DATA STORED IN CONTENT DATA STORAGE UNIT BASED ON CRYPTOGRAPHIC INFORMATION
S605
RECORD ENCRYPTED CONTENT DATA ON RECORDING MEDIUM 1720
END

fig. 15

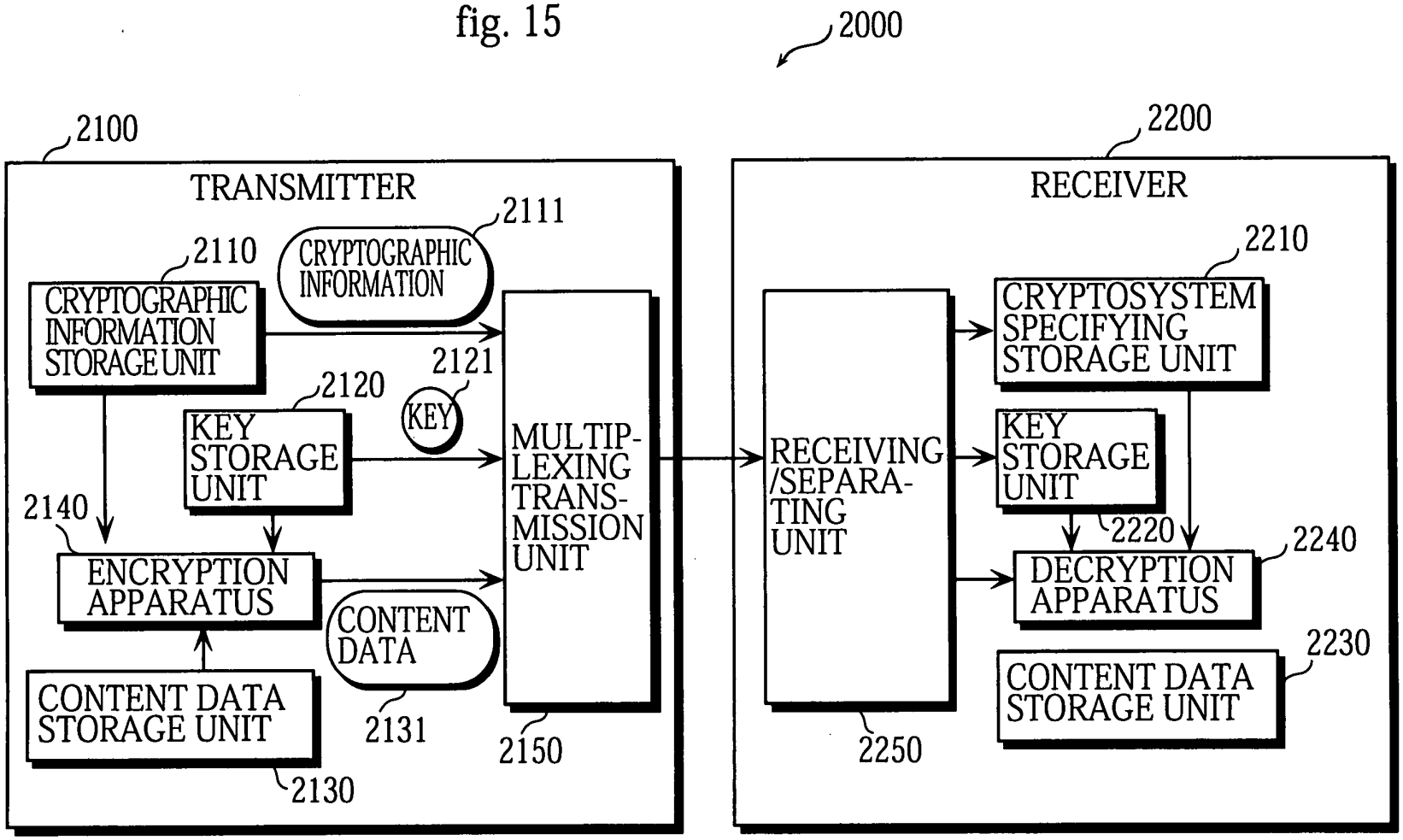

fig. 16

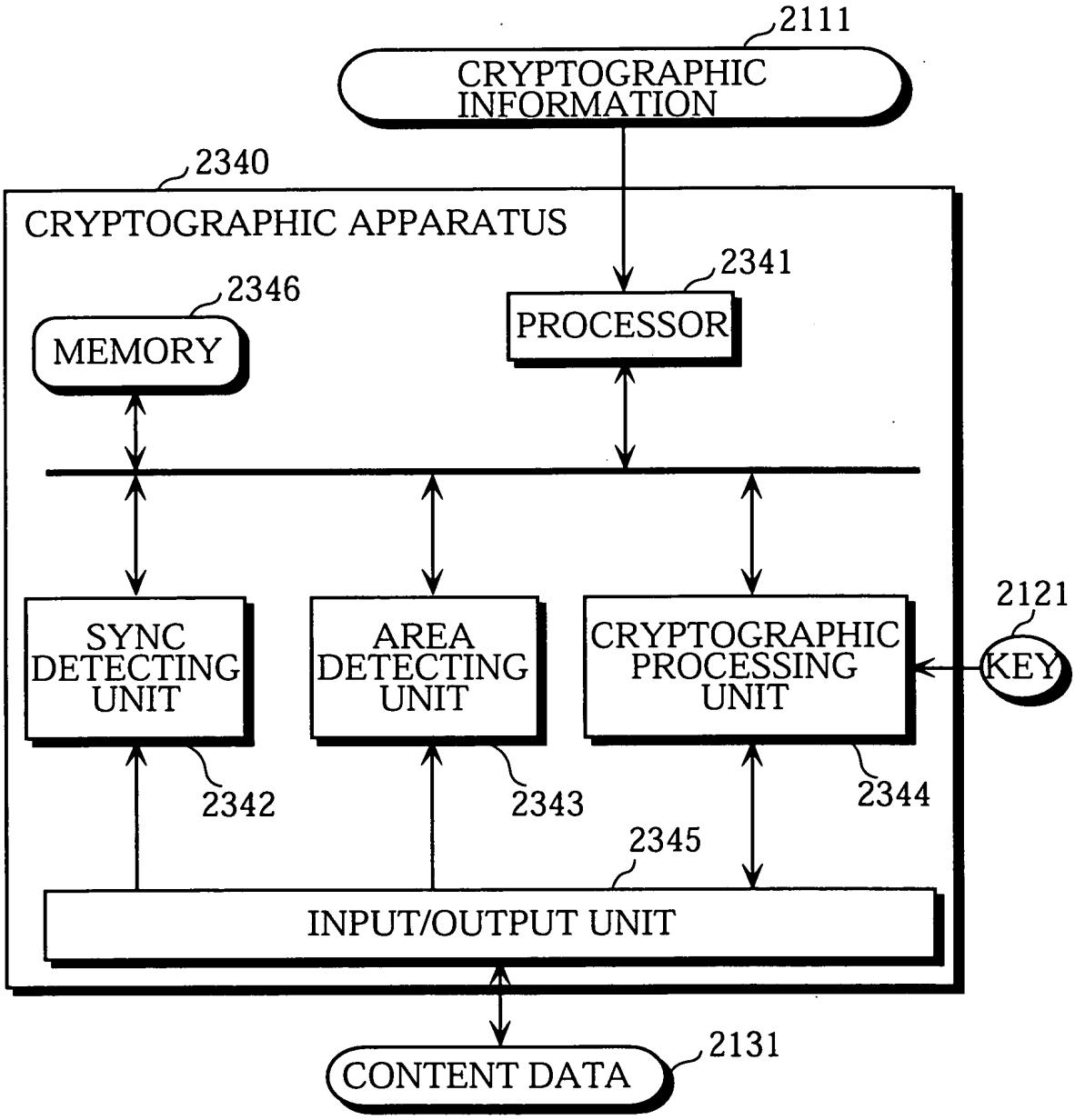
2111
CRYPTOGRAPHIC INFORMATION
2340
CRYPTOGRAPHIC APPARATUS
2341
PROCESSOR
2346
MEMORY
SYNC DETECTING UNIT
AREA DETECTING UNIT
CRYPTOGRAPHIC PROCESSING UNIT
2121
KEY
2342
2343
2344
2345
INPUT/OUTPUT UNIT
2131
CONTENT DATA

fig. 17

SYNC DETECTING INSTRUCTION Sync_detect ⟨END POINT⟩ ⟨REFERENCE ADDRESS⟩ ⟨SYNC DESCRIPTOR INFORMATION⟩
SYNC VERIFYING INSTRUCTION Sync_check ⟨END POINT⟩ ⟨REFERENCE ADDRESS⟩ ⟨SYNC DESCRIPTOR INFORMATION⟩
AREA DETECTING INSTRUCTION Area_detect ⟨POINT⟩ ⟨AREA DESCRIPTOR INFORMATION⟩
CRYPTOGRAPHIC PROCESSING INSTRUCTION Encrypt ⟨CRYPTOGRAPHY START POINT⟩ ⟨CRYPTOGRAPHY END POINT⟩ ⟨CRYPTOGRAPHIC DESCRIPTOR INFORMATION⟩

fig. 18

SYNC DESCRIPTOR INFORMATION

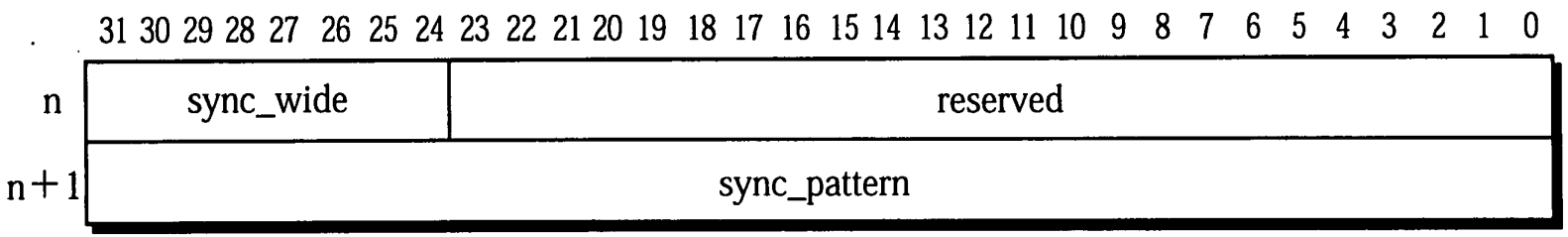

fig. 19

AREA DESCRIPTOR INFORMATION

	31	30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
n	as	flag_wide	reserved	flag_pattern1	flag_pattern2	flag_pnt
n+1	offset					
n+2	length_wide			length_type	length_unit	reserved	length_pnt

fig. 20

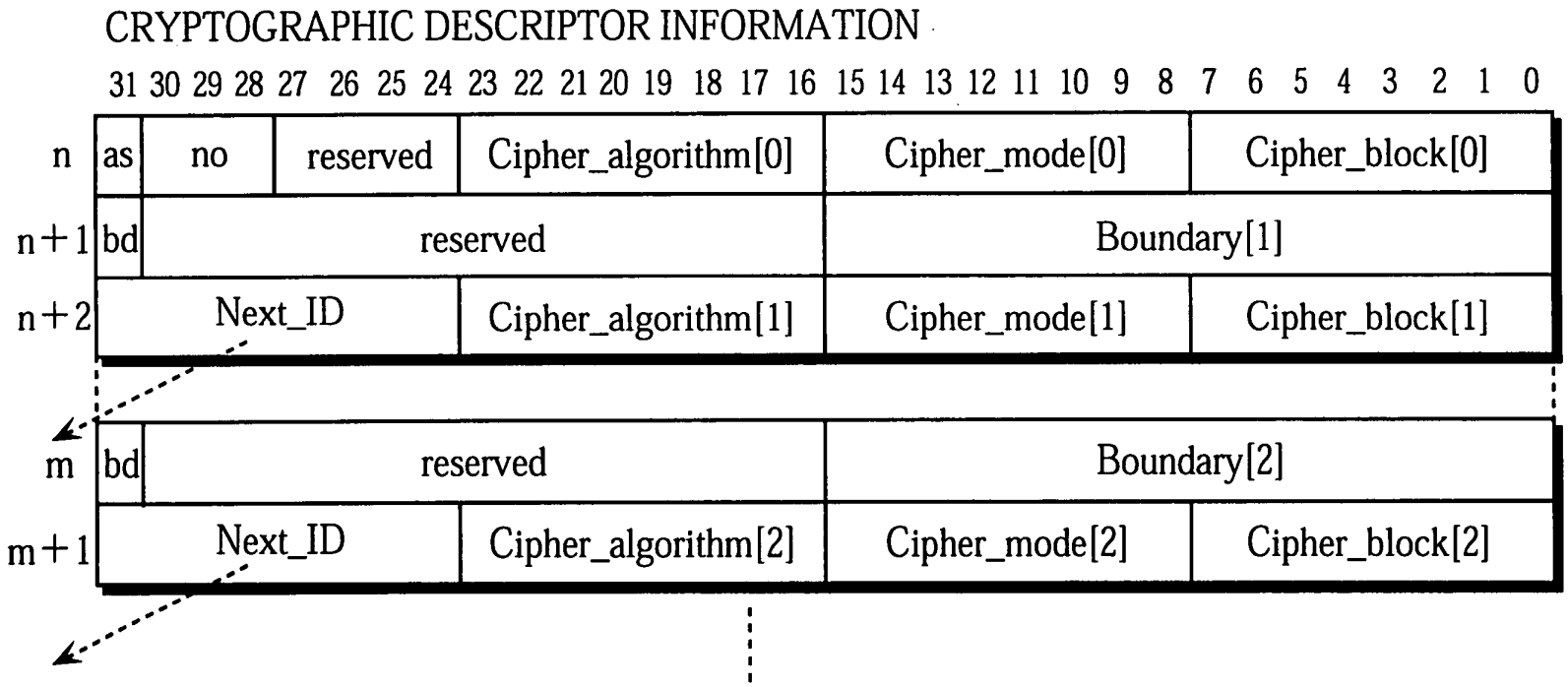

fig. 21

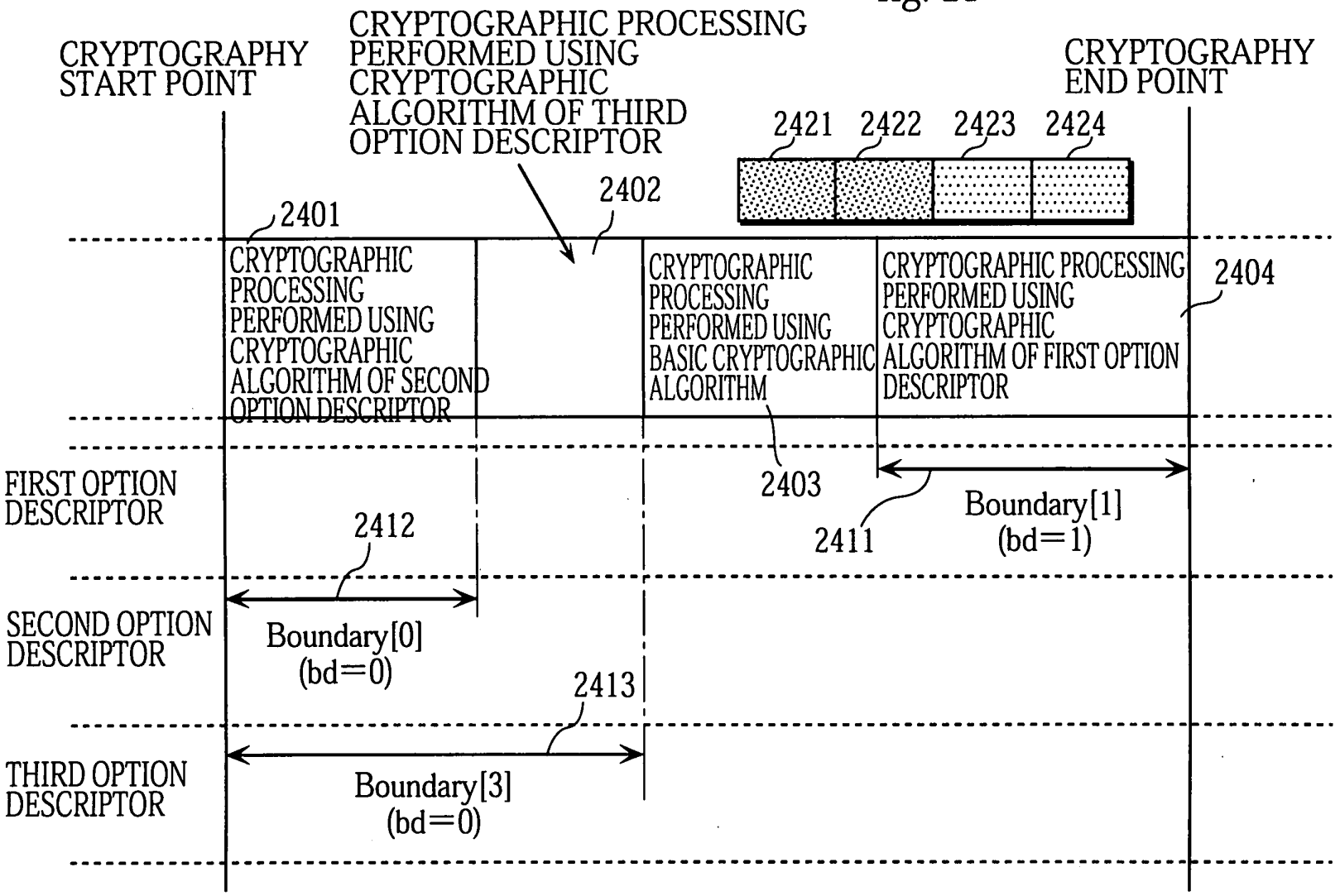

fig. 22

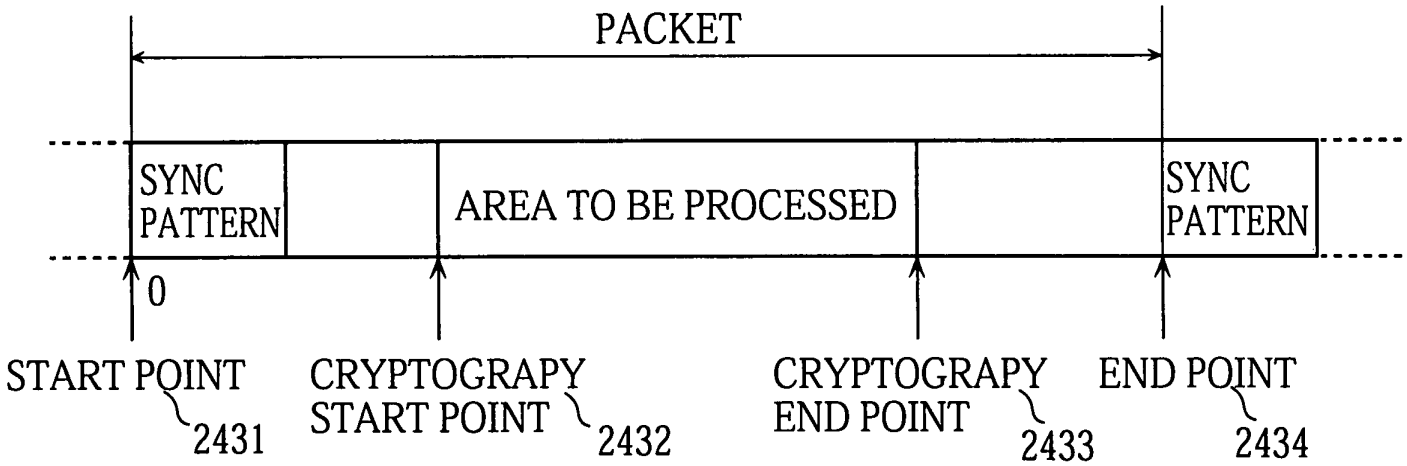

fig. 23

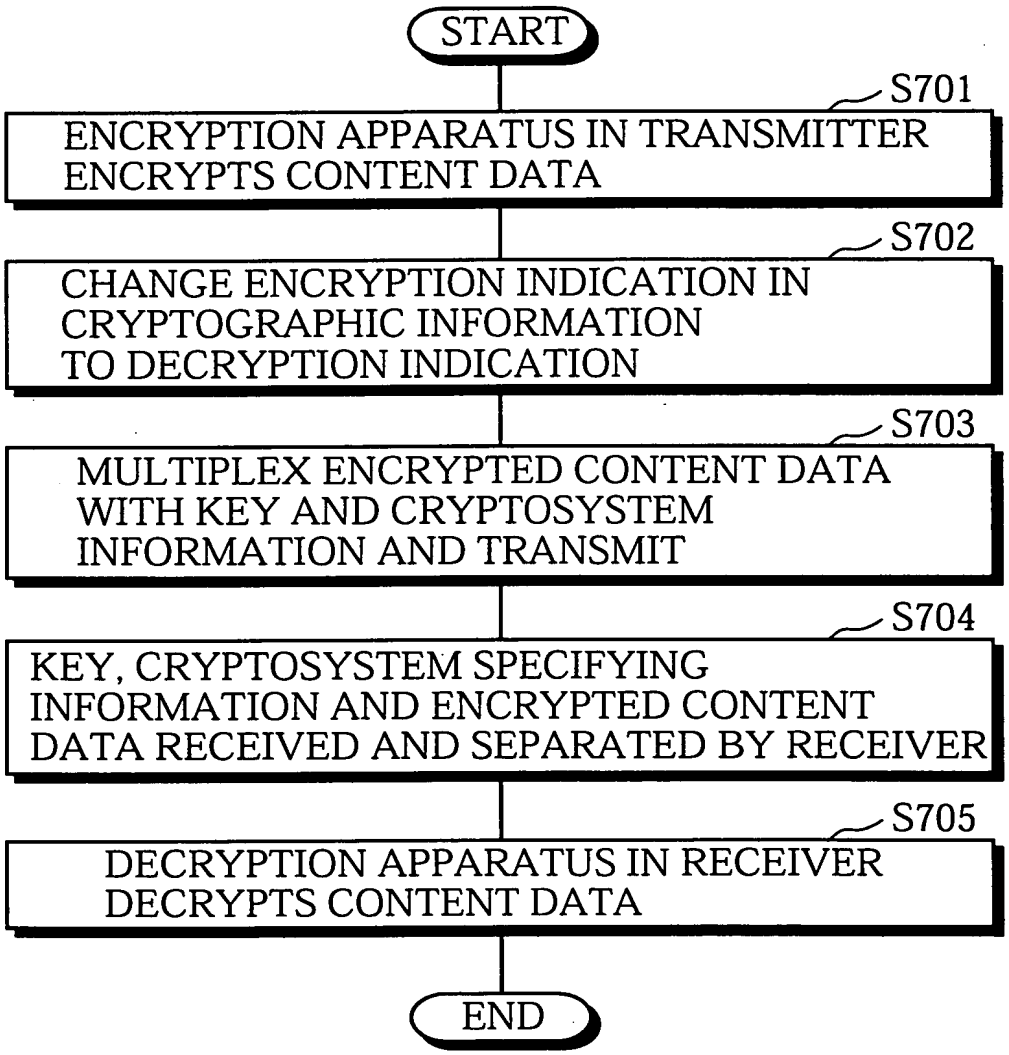
START
S701
ENCRYPTION APPARATUS IN TRANSMITTER ENCRYPTS CONTENT DATA
S702
CHANGE ENCRYPTION INDICATION IN CRYPTOGRAPHIC INFORMATION TO DECRYPTION INDICATION
S703
MULTIPLEX ENCRYPTED CONTENT DATA WITH KEY AND CRYPTOSYSTEM INFORMATION AND TRANSMIT
S704
KEY, CRYPTOSYSTEM SPECIFYING INFORMATION AND ENCRYPTED CONTENT DATA RECEIVED AND SEPARATED BY RECEIVER
S705
DECRYPTION APPARATUS IN RECEIVER DECRYPTS CONTENT DATA
END

fig. 24

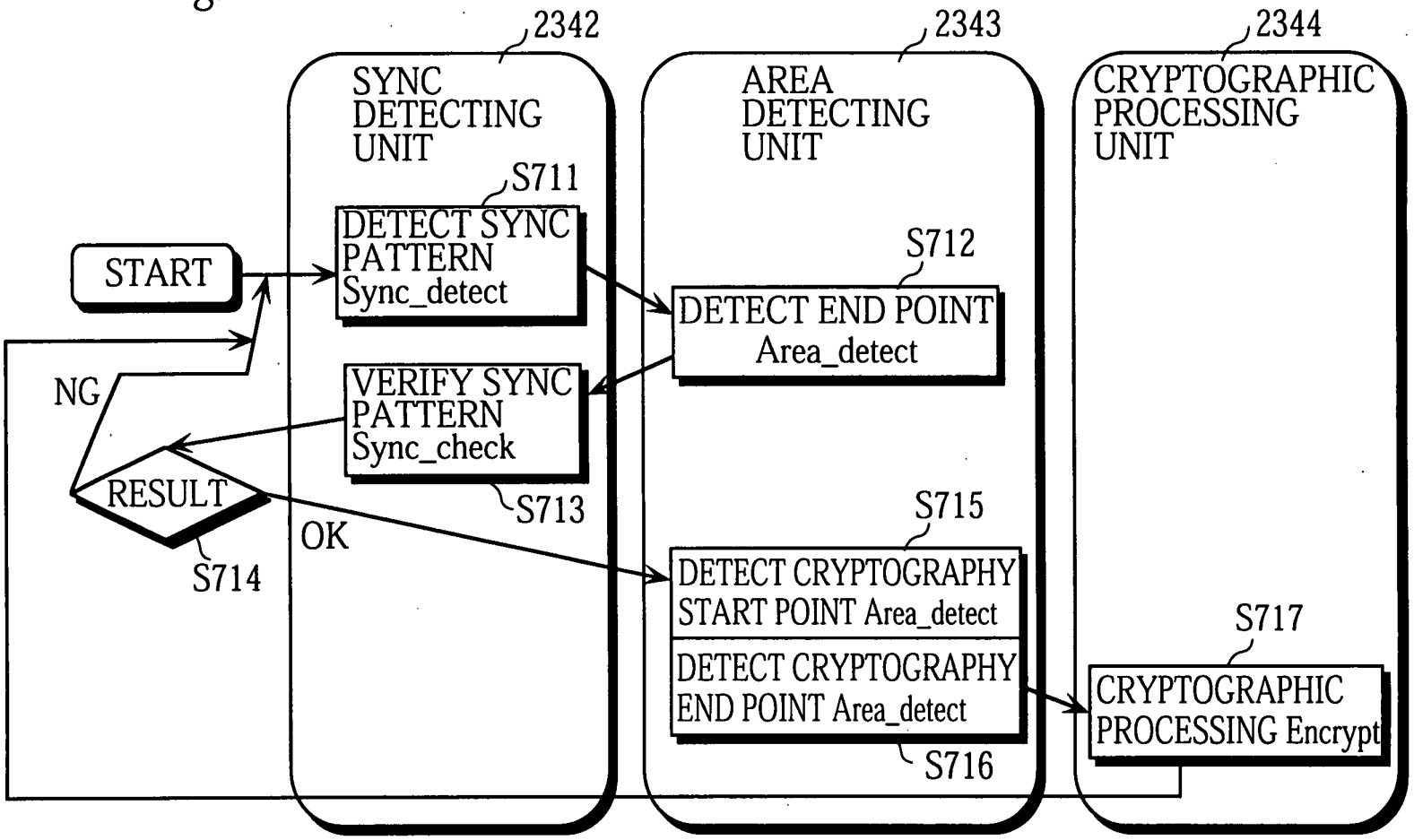
2342
SYNC DETECTING UNIT
2343
AREA DETECTING UNIT
2344
CRYPTOGRAPHIC PROCESSING UNIT
S711
DETECT SYNC PATTERN Sync_detect
START
S712
DETECT END POINT Area_detect
NG
VERIFY SYNC PATTERN Sync_check
RESULT
S713
S714
OK
S715
DETECT CRYPTOGRAPHY START POINT Area_detect
DETECT CRYPTOGRAPHY END POINT Area_detect
S716
S717
CRYPTOGRAPHIC PROCESSING Encrypt

fig. 25

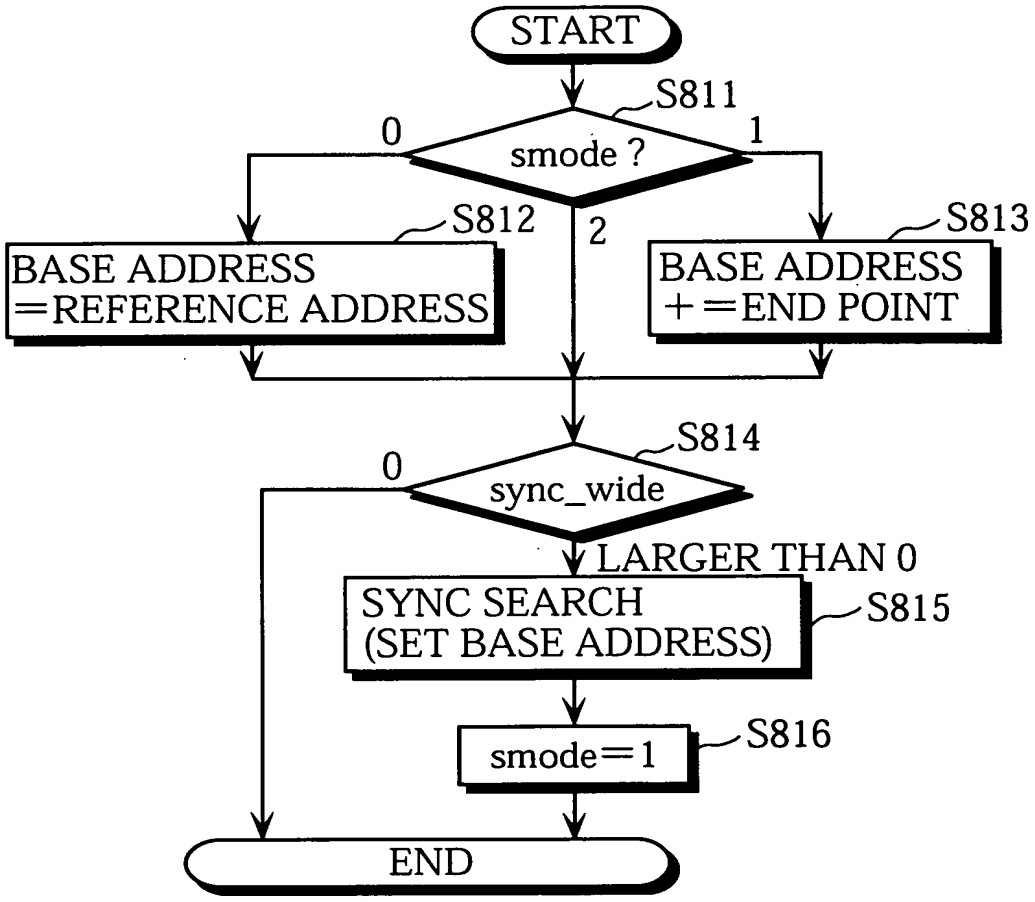
START
S811
smode ?
0
1
2
S812
BASE ADDRESS
=REFERENCE ADDRESS
S813
BASE ADDRESS
+=END POINT
S814
sync_wide
0
LARGER THAN 0
SYNC SEARCH
(SET BASE ADDRESS)
S815
smode=1
S816
END

fig. 26

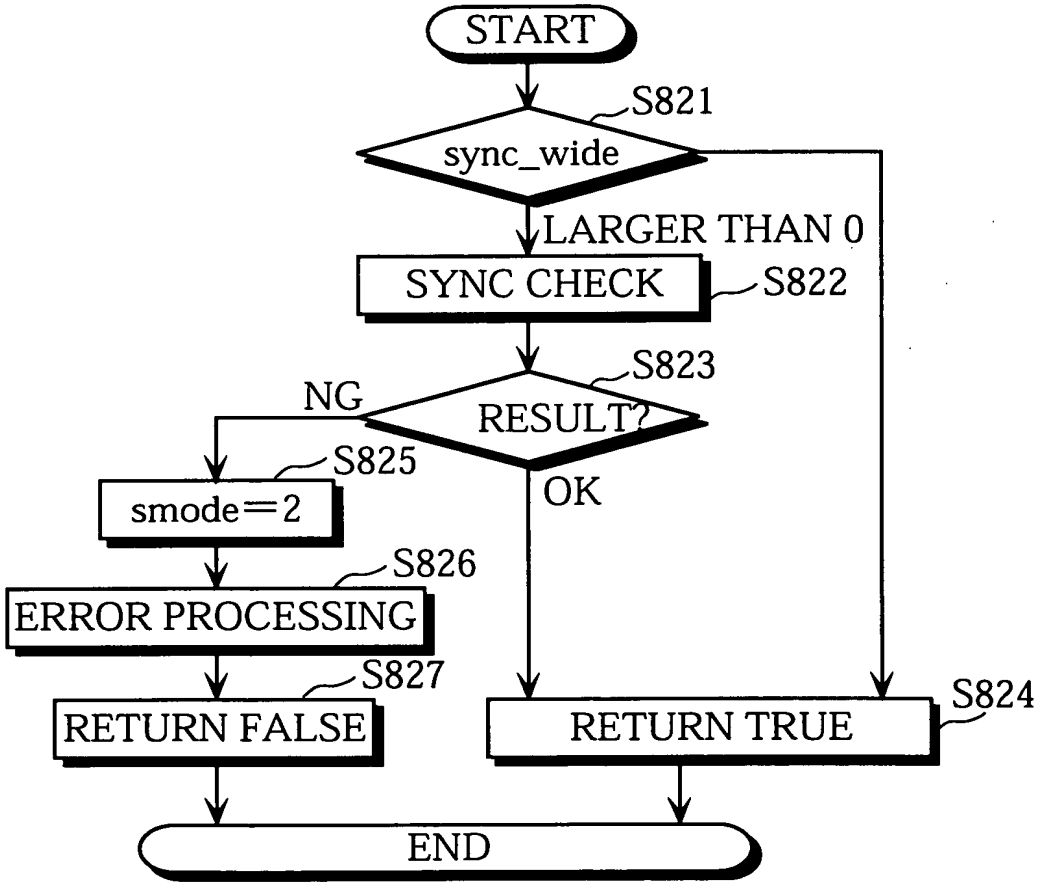
START
S821
sync_wide
LARGER THAN 0
SYNC CHECK
S822
S823
NG
RESULT?
OK
S825
smode=2
S826
ERROR PROCESSING
S827
RETURN FALSE
RETURN TRUE
S824
END

fig. 27

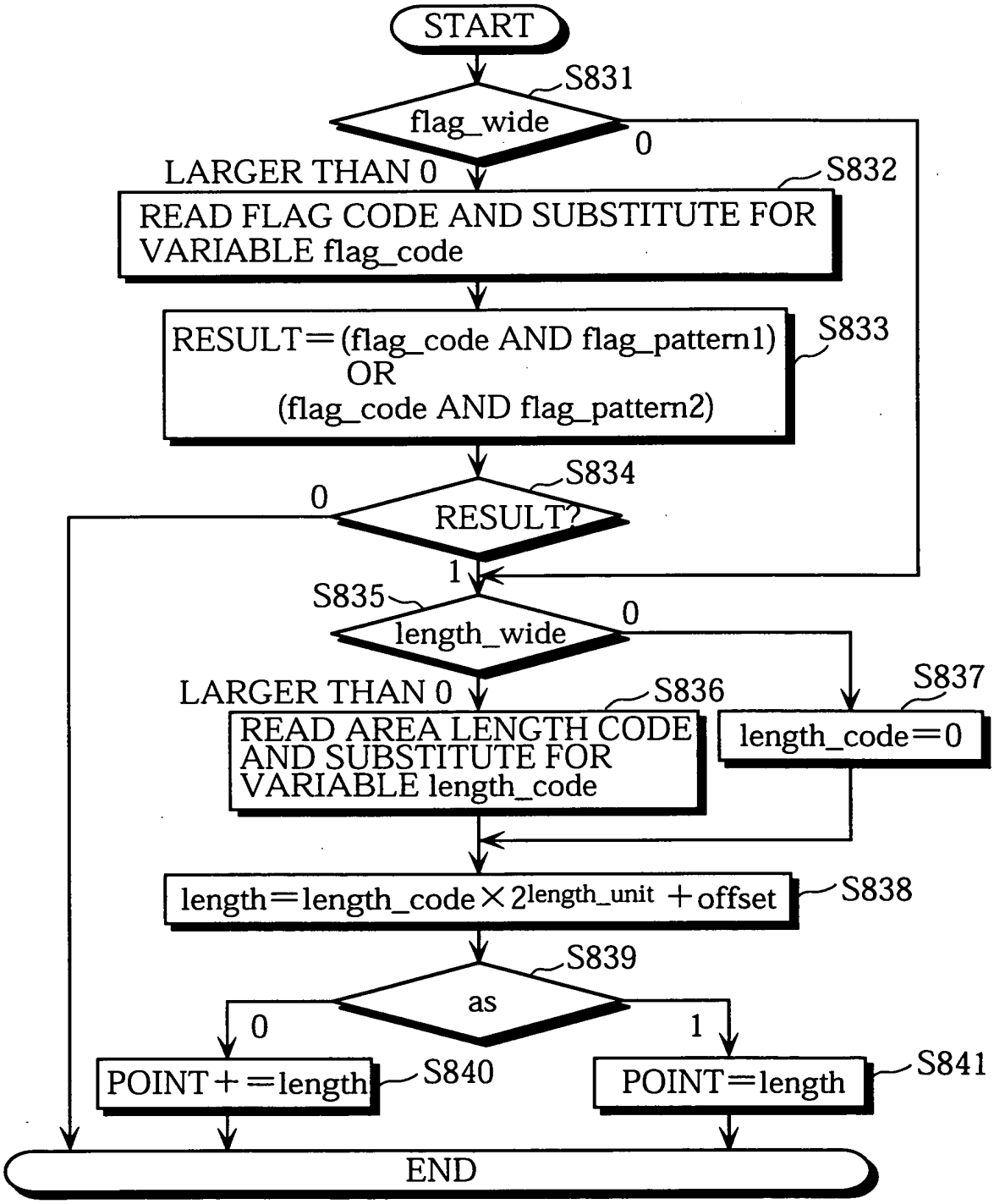
START
S831
flag_wide
0
LARGER THAN 0
S832
READ FLAG CODE AND SUBSTITUTE FOR VARIABLE flag_code
RESULT=(flag_code AND flag_pattern1) OR (flag_code AND flag_pattern2)
S833
S834
0
RESULT?
1
S835
length_wide
0
LARGER THAN 0
S836
READ AREA LENGTH CODE AND SUBSTITUTE FOR VARIABLE length_code
S837
length_code=0
length=length_code×2^length_unit +offset
S838
S839
as
0
1
POINT+=length
S840
POINT=length
S841
END

fig. 28

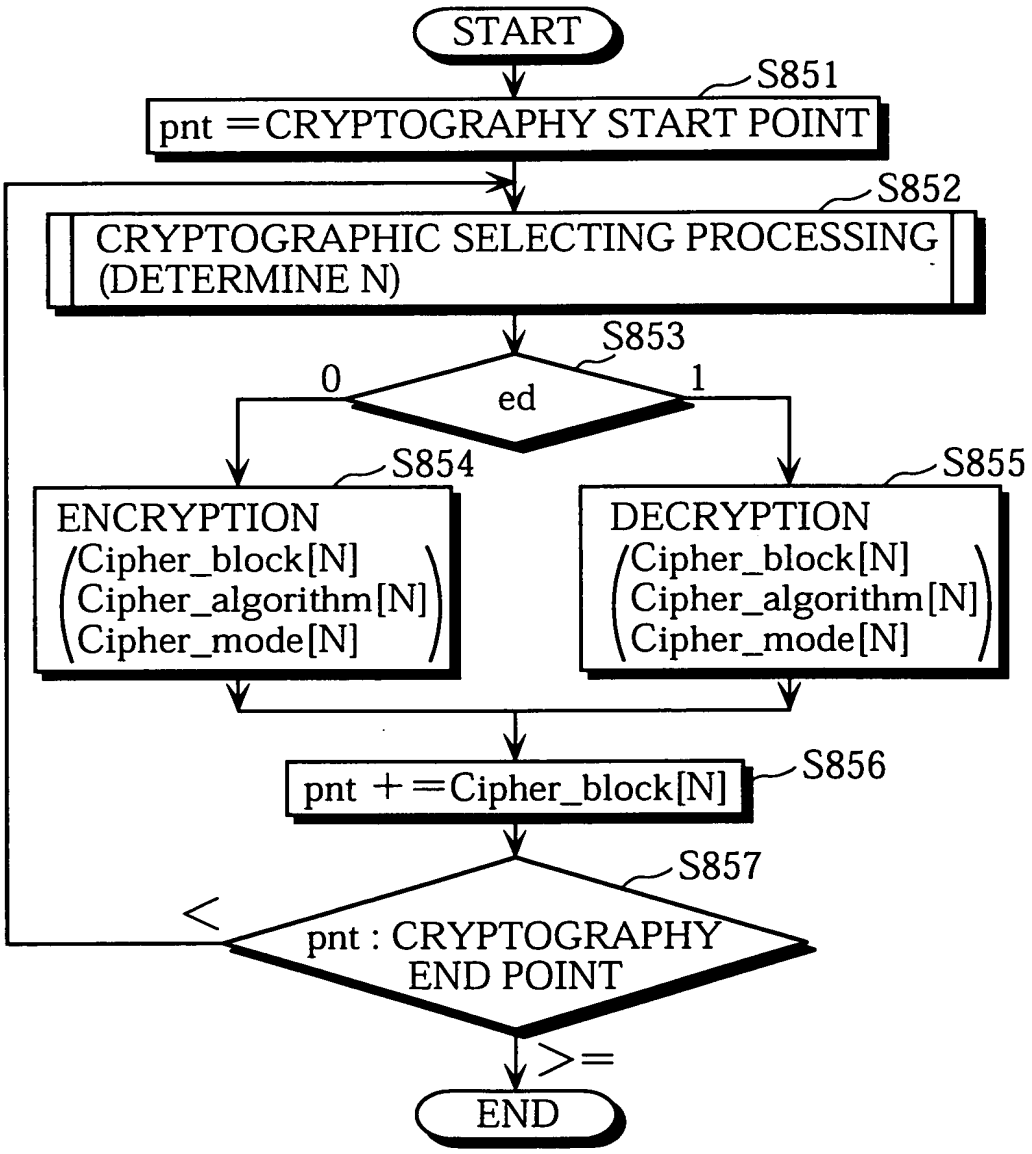
START
S851
pnt =CRYPTOGRAPHY START POINT
S852
CRYPTOGRAPHIC SELECTING PROCESSING
(DETERMINE N)
S853
ed
0
1
S854
ENCRYPTION
(Cipher_block[N]
Cipher_algorithm[N]
Cipher_mode[N]
S855
DECRYPTION
(Cipher_block[N]
Cipher_algorithm[N]
Cipher_mode[N]
pnt +=Cipher_block[N]
S856
S857
pnt : CRYPTOGRAPHY
END POINT
<
>=
END

fig. 29

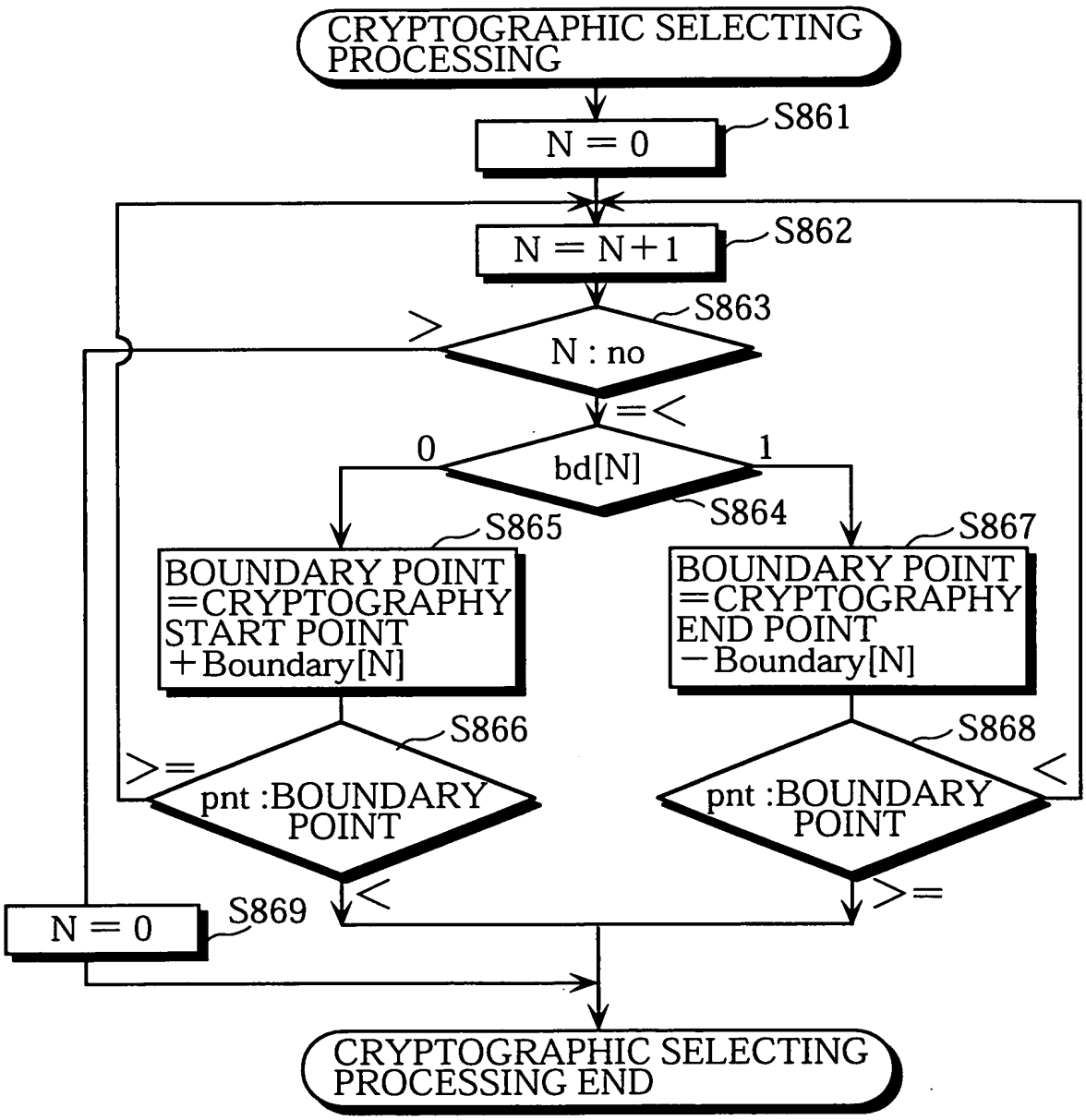
CRYPTOGRAPHIC SELECTING PROCESSING
N = 0
S861
N = N+1
S862
S863
N : no
>
=<
0
bd[N]
1
S864
S865
BOUNDARY POINT =CRYPTOGRAPHY START POINT +Boundary[N]
S867
BOUNDARY POINT =CRYPTOGRAPHY END POINT −Boundary[N]
S866
>=
pnt :BOUNDARY POINT
<
S868
pnt :BOUNDARY POINT
<
>=
N = 0
S869
CRYPTOGRAPHIC SELECTING PROCESSING END

fig. 30

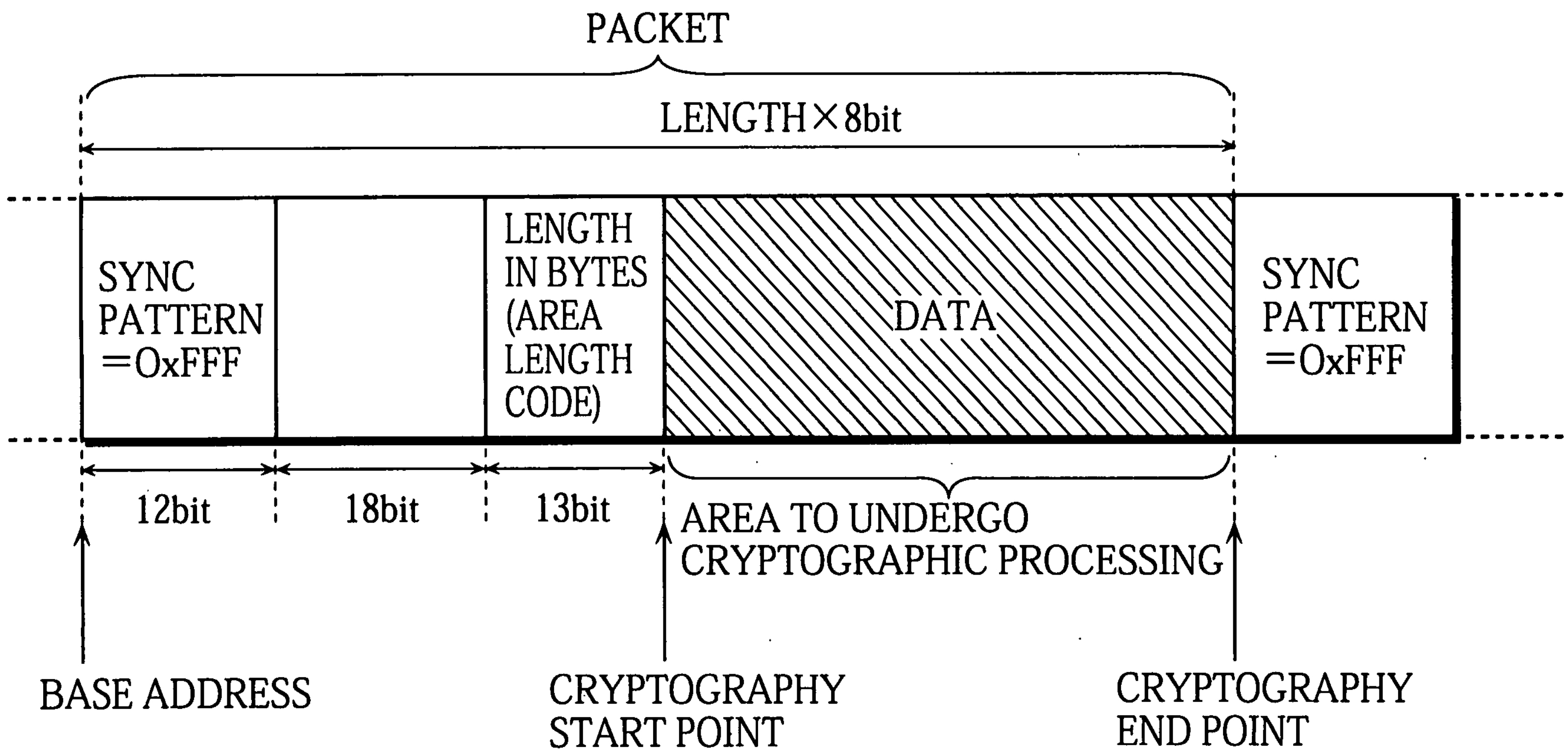
PACKET
LENGTH×8bit
SYNC PATTERN =0xFFF
LENGTH IN BYTES (AREA LENGTH CODE)
DATA
SYNC PATTERN =0xFFF
12bit
18bit
13bit
AREA TO UNDERGO CRYPTOGRAPHIC PROCESSING
BASE ADDRESS
CRYPTOGRAPHY START POINT
CRYPTOGRAPHY END POINT

fig. 31